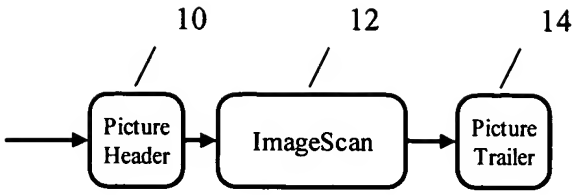


FIG. 1

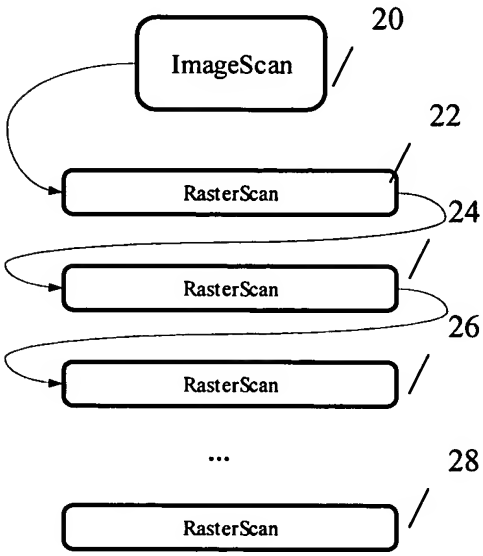


FIG. 2

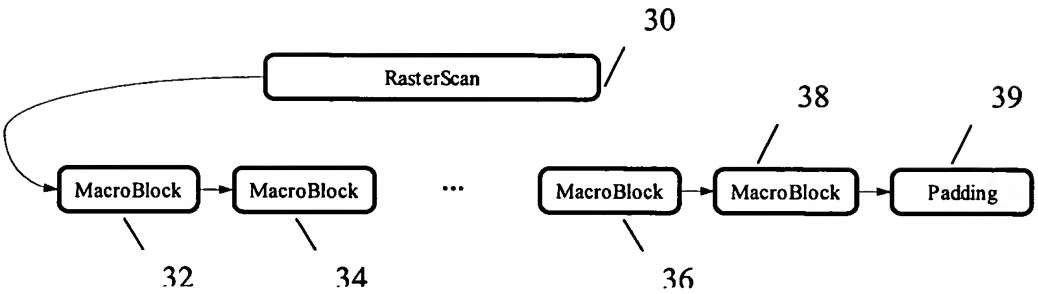
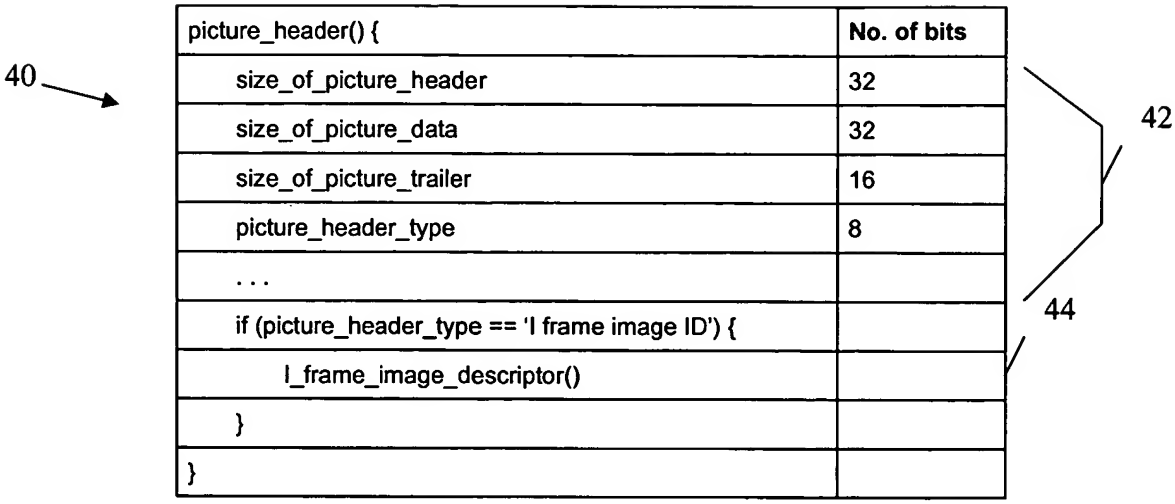


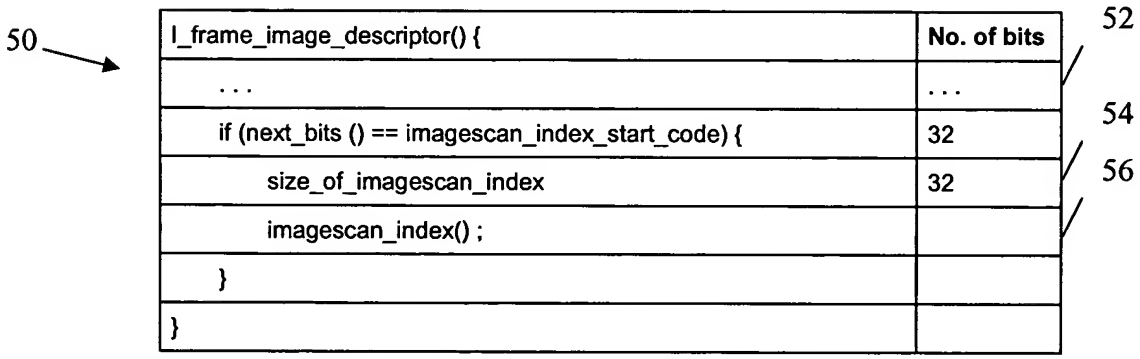
FIG. 3

FIG. 4



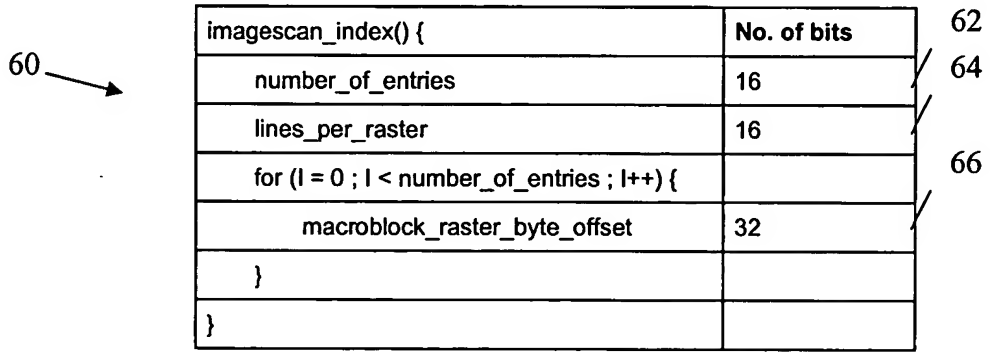
picture_header() {	No. of bits
size_of_picture_header	32
size_of_picture_data	32
size_of_picture_trailer	16
picture_header_type	8
. . .	
if (picture_header_type == 'I frame image ID') {	
I_frame_image_descriptor()	
}	
}	

FIG. 5



I_frame_image_descriptor() {	No. of bits
. . .	. . .
if (next_bits () == imagescan_index_start_code) {	32
size_of_imagescan_index	32
imagescan_index() ;	
}	
}	

FIG. 6



imagescan_index() {	No. of bits
number_of_entries	16
lines_per_raster	16
for (I = 0 ; I < number_of_entries ; I++) {	
macroblock_raster_byte_offset	32
}	
}	

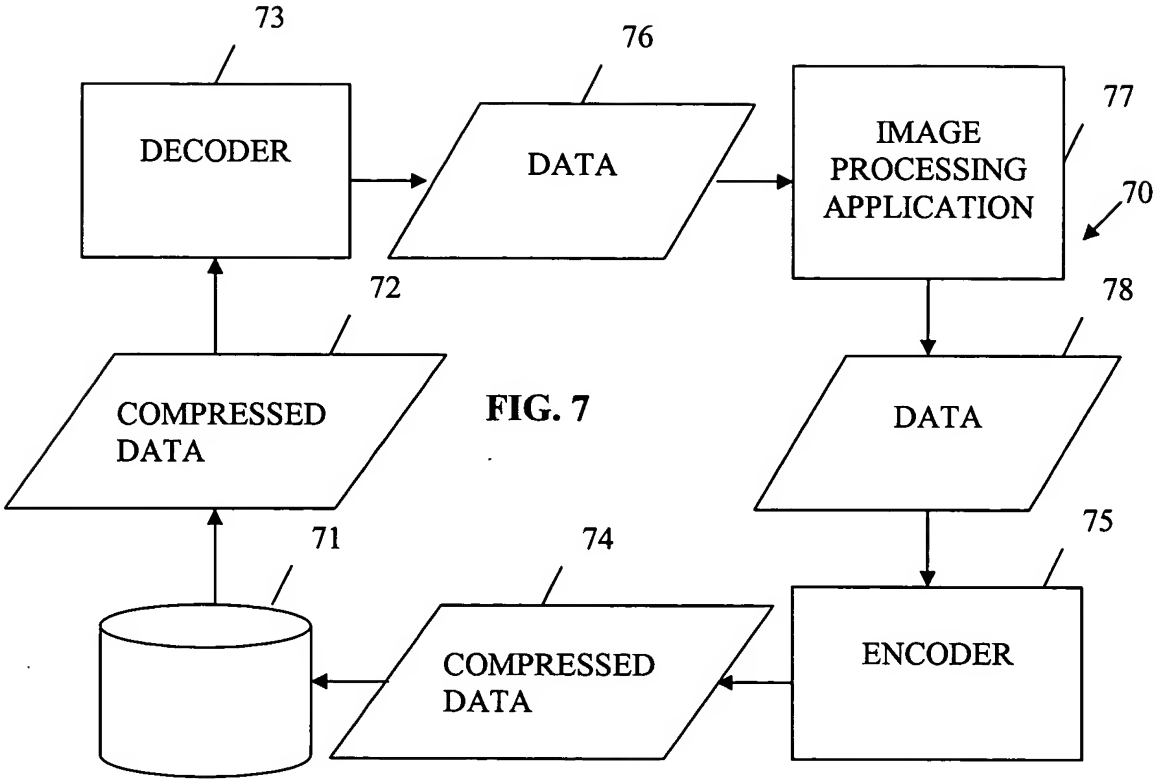

73
DECODER
76
DATA
77
IMAGE PROCESSING APPLICATION
70
72
COMPRESSED DATA
78
DATA
71
74
COMPRESSED DATA
75
ENCODER

FIG. 7

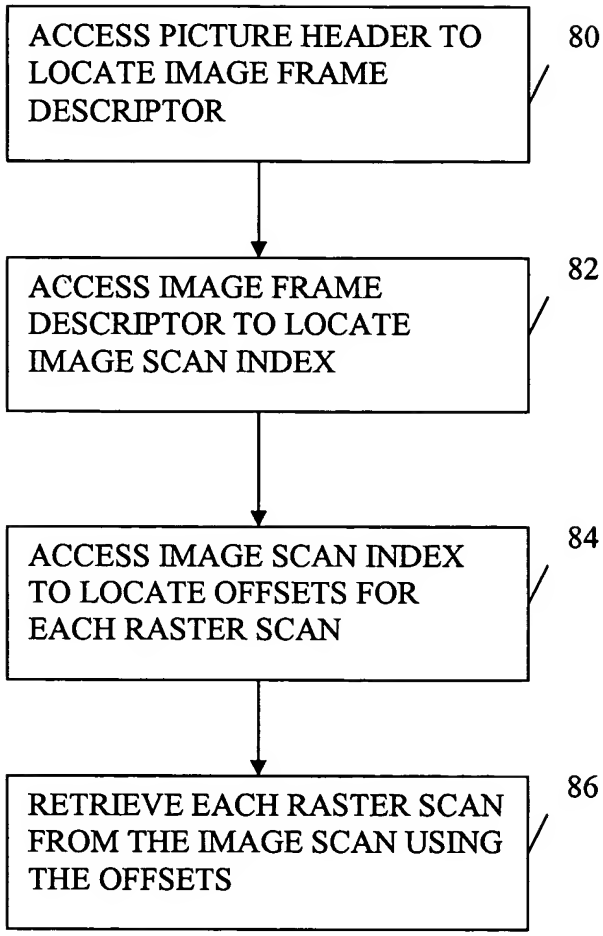

ACCESS PICTURE HEADER TO LOCATE IMAGE FRAME DESCRIPTOR
80
ACCESS IMAGE FRAME DESCRIPTOR TO LOCATE IMAGE SCAN INDEX
82
ACCESS IMAGE SCAN INDEX TO LOCATE OFFSETS FOR EACH RASTER SCAN
84
RETRIEVE EACH RASTER SCAN FROM THE IMAGE SCAN USING THE OFFSETS
86

FIG. 8

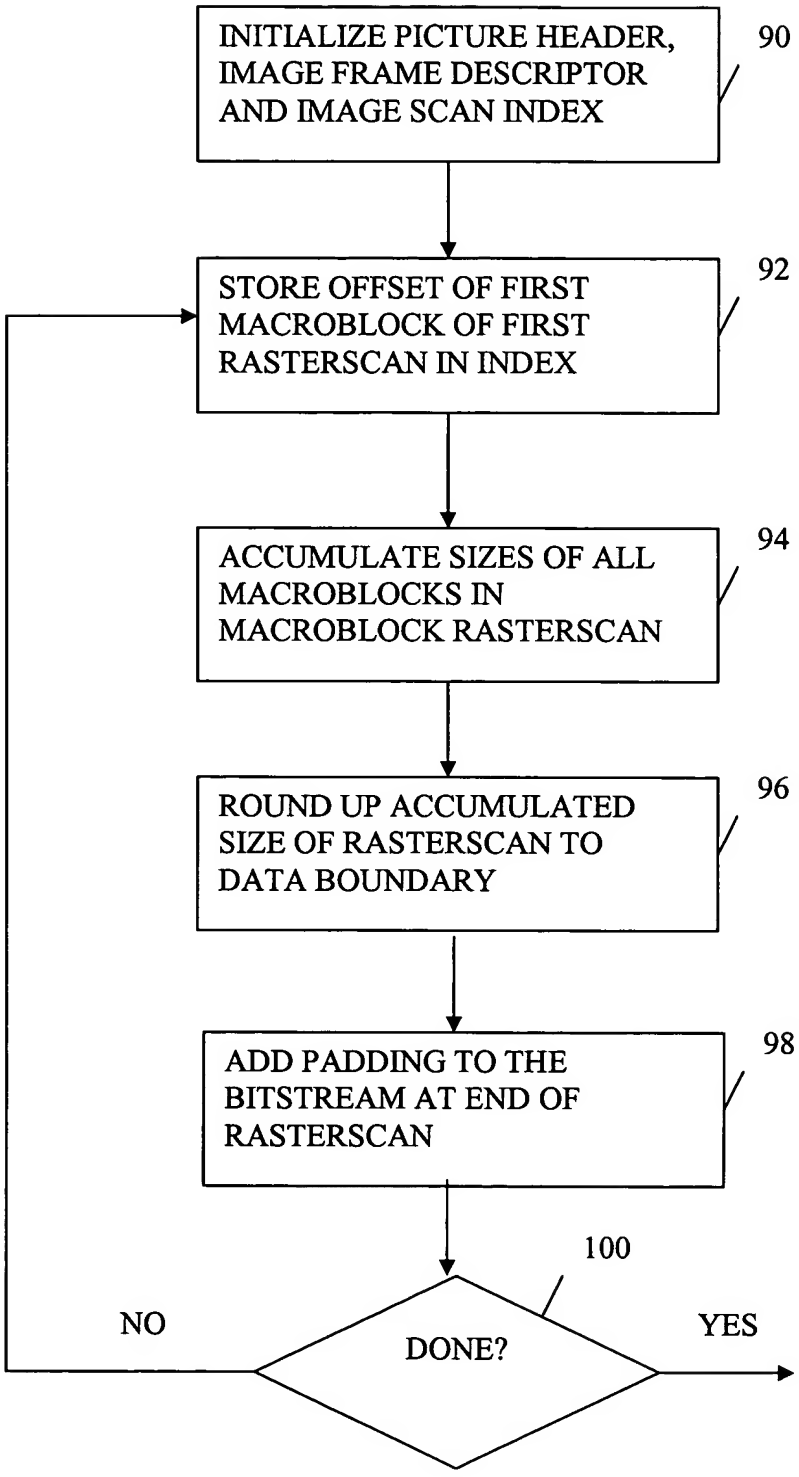

INITIALIZE PICTURE HEADER, IMAGE FRAME DESCRIPTOR AND IMAGE SCAN INDEX
90
STORE OFFSET OF FIRST MACROBLOCK OF FIRST RASTERSCAN IN INDEX
92
ACCUMULATE SIZES OF ALL MACROBLOCKS IN MACROBLOCK RASTERSCAN
94
ROUND UP ACCUMULATED SIZE OF RASTERSCAN TO DATA BOUNDARY
96
ADD PADDING TO THE BITSTREAM AT END OF RASTERSCAN
98
100
DONE?
NO
YES

FIG. 9